Kestrel

Student: Ved Thiru
Mentor: Paul Coccoli

Georgia Tech
Open Source Program Office

Project Overview

Threat hunters develop intrusion detection systems against cybersecurity threats.

Before Kestrel: Hunters need to write complex queries in multiple query languages to query all their data sources, primarily system and network activities.

Many queries are written for specific data sources and types, so you can't reuse the same query for a different data source.

After Kestrel: Threat hunters write their queries in Kestrel, which compiles to various other query languages and APIs. Hunters can reuse steps and analytics across different data sources.

Kestrel helps with the specific implementation of the query against a particular data source, so that hunters can focus on the logic.

Kestrel also assembles the results into entities for easier analysis.

Goals and Milestones

Milestone 1: Implement the necessary changes in the frontend for the sequence syntax. Specifically, this is support for the syntax `A FOLLOWED BY B WITHIN N SECONDS SAME host` in the parsing module, and adding necessary `Instruction`s, etc.

Milestone 2: Implement the IR and graph changes necessary for querying sequences.

Highlights and Accomplishments

I implemented support for querying sequences: "Find an event of one type, followed by an event of another type."

To do this, I

- added new instructions to the IR
- modified the graph (since sequences are the first feature in Kestrel to hold three references)
- added SQL generation for sequencing
- modified filters so that they are applied after sequences
- cached results of sequencing, because it is an expensive operation in SQL
- wrote some documentation!
- wrote many tests!

Open Source Outcomes

Generally, I learned how to start working on a project with

- an existing codebase
- with not much internal documentation.

I also learned the importance of **thorough tests**. This let me tinker around and learn the codebase, without having to fear messing up. Especially critical if there's not much documentation.

Future Work

The easiest way to continue my work, specifically, would be to implement sequencing support for OpenSearch (right now it only generates SQL).

Adding support for chained sequences (`A FOLLOWED BY B FOLLOWED BY C WITHIN N SECS SAME host`) would also be a good idea.

Georgia Tech